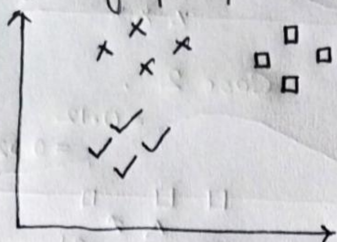
t-SNE (t-distributed stochastic neighbor embedding)

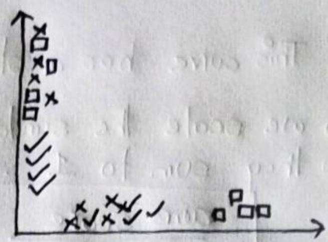
→ It takes the high dimensional dataset & reduce it to low dimensional graph. which retains lots of the information.

→ suppose we take a graph of 2 dimension data points.

2D scatter plot



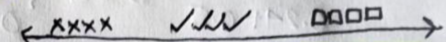
directly plotting on X and y axis directly



(seperately)

Even if we plot all the datapoint on X or Y axis, we see a mishmash (no separable clusters)

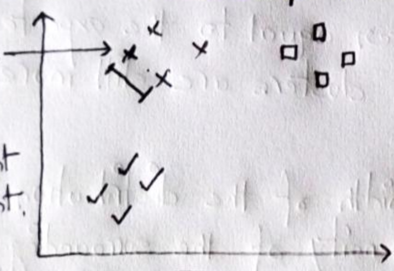
- So what t-SNE does it find a way to project data into a low dimensional space (in this case 1D number line) so that clustering in the high dimensional space (in this case, the 2D scatter plot) is preserved.



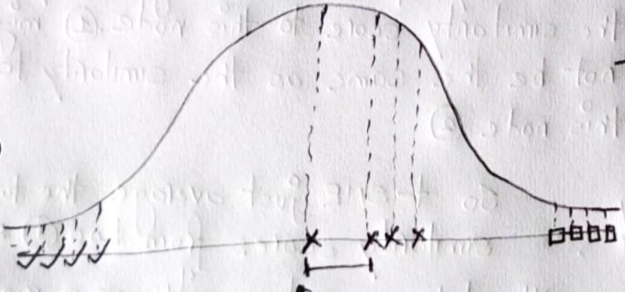
t-SNE 1D number line.

Step 1 – Determine the similarity of all the points in scatter plot.

For this example let's determine the similarities between the point and all other point.



First measure the distance between point. Then plot that distance on a normal curve that is centered on the point of interest and then draw a line from point of curve. The length of that line is the "unscaled similarity".



→ Using a normal distribution means the distant points have very low similarity value and near points have very high similarity value.

→ Measure the distances from the points to the curve to get the unscaled similarity score with respect to the point of interest.

This point (from which we will measure) will be in middle of distribution

step 2

→ Next step, is to scale the unscaled similarities so that they add up to 1.

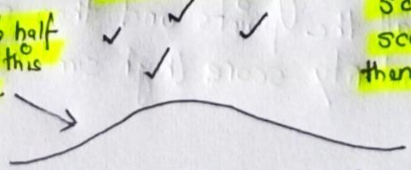
So why do the similarity scores need to add up to 1?

- Width of the normal curve depend on the density of data near point of interest.

suppose we have high density :

This curve is half as wide as this curve

Low density.

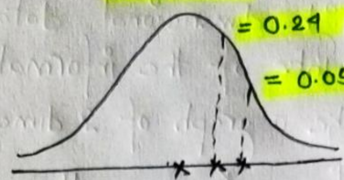


Scaling the similarity scores will make them same for both clusters.

Here an example,

This curve has a standard deviation = 1

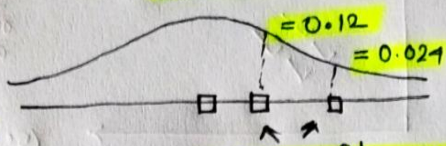
Case 1



The "unscaled" similarity values.

This curve has a standard deviation = 2.

Case 2.



The "unscaled" similarity values are half of others one.

These points are twice as far from middle.

So we scale the similarity score so they sum to 1.

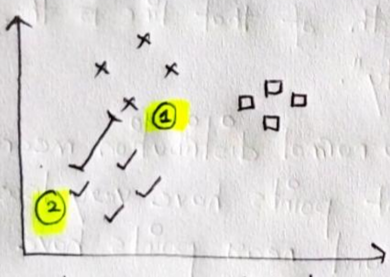
$$\boxed{\frac{\text{Score}}{\text{Sum of all scores}} = \text{Scaled Score}}$$

Case 1, $\frac{0.24}{0.24+0.5} = 0.82$, $\frac{0.05}{0.24+0.5} = 0.18$.

Case 2, $\frac{0.12}{0.12+0.024} = 0.82$, $\frac{0.024}{0.12+0.024} = 0.18$.

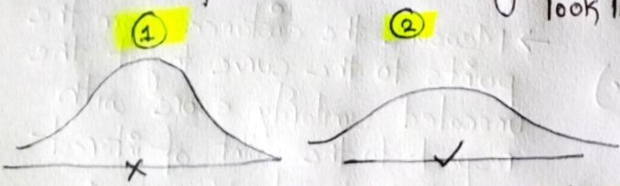
If we see proportion are same (0.82, 0.18). This implies that the scaled similarity scores for this relatively tight clusters (x) and the relatively loose clusters (□) are same.

→ t-SNE has a "perplexity" parameter equal to the expected density and that comes into play, but these clusters are still more "similar" than we might expect.



- The width of the distribution is based on the density of the surrounding data points, the similarity score to this node ① might not be the same as the similarity to this node ②

Due to density, two distribution graph may look like:



So t-SNE just average the two similarity scores from two ① & ②

Ultimately, we end up with a matrix of similarity score.

- The distribution which we use here is "t-distribution" (lot like a normal distribution) except the "t" isn't as tall in the middle and tails are taller on the ends.

The "t-distribution" is the "t" in t-SNE.

- Summary → So using a t-distribution we calculate "unscaled" similarity score for all the points and then scale them. Then we end with a matrix of similarity score (high similarity & low similarity).